

KEKER & VAN NEST LLP
 ROBERT A. VAN NEST - #84065
 rvannest@kvn.com
 CHRISTA M. ANDERSON - #184325
 canderson@kvn.com
 633 Battery Street
 San Francisco, CA 94111-1809
 Telephone: 415.391.5400
 Facsimile: 415.397.7188

KING & SPALDING LLP
 DONALD F. ZIMMER, JR. - #112279
 fzimmer@kslaw.com
 CHERYL A. SABNIS - #224323
 csabnis@kslaw.com
 101 Second St., Suite 2300
 San Francisco, CA 94105
 Telephone: 415.318.1200
 Facsimile: 415.318.1300

KING & SPALDING LLP
 SCOTT T. WEINGAERTNER (*Pro Hac Vice*)
 sweingaertner@kslaw.com
 ROBERT F. PERRY
 rperry@kslaw.com
 BRUCE W. BABER (*Pro Hac Vice*)
 1185 Avenue of the Americas
 New York, NY 10036
 Telephone: 212.556.2100
 Facsimile: 212.556.2222

GREENBERG TRAURIG, LLP
 IAN C. BALLON - #141819
 ballon@gtlaw.com
 HEATHER MEEKER - #172148
 meekerh@gtlaw.com
 1900 University Avenue
 East Palo Alto, CA 94303
 Telephone: 650.328.8500
 Facsimile: 650.328.8508

Attorneys for Defendant
 GOOGLE INC.

UNITED STATES DISTRICT COURT
 NORTHERN DISTRICT OF CALIFORNIA
 SAN FRANCISCO DIVISION

ORACLE AMERICA, INC.,

 Plaintiff,

 v.

 GOOGLE INC.,

 Defendant.

Case No. 3:10-cv-03561-WHA

**GOOGLE INC.'S REPLY IN SUPPORT
 OF MOTION FOR SUMMARY
 JUDGMENT ON COUNT VIII OF
 PLAINTIFF ORACLE AMERICA'S
 AMENDED COMPLAINT**

Judge: Hon. William Alsup

Hearing: 2:00 p.m., September 15, 2011

TABLE OF CONTENTS

	<u>Page</u>
I. INTRODUCTION	1
II. ARGUMENT	1
A. The Court should ignore or strike Oracle’s improper objections.	1
B. Oracle must prove “virtual identity” between the works, which it cannot do.	1
C. The Java language APIs are methods of operation.	2
1. Oracle repeatedly mischaracterizes the APIs in an attempt to show creative expression where there is none.	2
2. Oracle’s examples of “creativity” in the design of APIs are examples of form following function, and thus militate <i>against</i> copyrightability.	3
a. The selection of elements to include in a set of APIs is not copyrightable.	4
b. The structure for a set of APIs is not copyrightable.	5
c. The names of the elements in the APIs are not copyrightable.	6
3. Oracle improperly relies on dated law and other inapposite authorities.	6
D. Any creative expression in the Java language APIs has merged into their underlying ideas, or is unprotected <i>scenes a faire</i>	8
E. Google’s fair use case is <i>stronger</i> than that of <i>Sony v. Connectix</i> , in which the Ninth Circuit held that Connectix’s use was fair.	9
F. Oracle has identified no inconsistency between Google’s position and Google’s treatment of its own APIs.	10
G. Oracle offers no evidence of substantial similarity between the documentation of the Android core libraries and the Java language API documentation.	11
H. The alleged copying in portions of 12 files out of several thousands of files is <i>de minimis</i> and thus not actionable.	12
1. The alleged copying must be considered against the entirety of the Asserted Works to determine whether it is <i>de minimis</i>	12
2. The rangeCheck function is not qualitatively or quantitatively significant.	13

TABLE OF CONTENTS
(cont'd)

		<u>Page</u>
3.	The eight allegedly decompiled files are not qualitatively or quantitatively significant.....	14
4.	The allegedly copied comments in the remaining two files are not qualitatively or quantitatively significant.	14
5.	Google's implementation of 37 Java language API packages is irrelevant to the <i>de minimis</i> analysis.....	15
III.	CONCLUSION.....	15

TABLE OF AUTHORITIES

Page(s)**Federal Cases**

<i>Allen v. Academic Games League of America, Inc.</i> 89 F.3d 614 (9th Cir. 1996)	7
<i>Apple Computer, Inc. v. Formula, Int'l, Inc.</i> 725 F.2d 521 (9th Cir. 1984)	7
<i>Apple Computer, Inc. v. Microsoft Corp.</i> 35 F.3d 1435 (9th Cir. 1994)	1, 8
<i>ATC Distribution Group, Inc. v. Whatever It Takes Transmissions & Parts, Inc.</i> 402 F.3d 700 (6th Cir. 2005)	3
<i>Autoskill Inc. v. National Educational Support Systems, Inc.</i> 994 F.2d 1476 (10th Cir. 1993)	8
<i>Bean v. McDougal Littell</i> 669 F. Supp. 2d 1031 (D. Ariz. 2008)	12
<i>Brown Bag Software v. Symantec Corp.</i> 960 F.2d 1465 (9th Cir. 1992)	7, 8
<i>CMAX/Cleveland, Inc. v. UCR, Inc.</i> 804 F. Supp. 337 (M.D. Ga. 1992)	8
<i>Engineering Dynamics, Inc. v. Structural Software, Inc.</i> 26 F.3d 1335 (5th Cir. 1994)	8
<i>Feist Pubs., Inc. v. Rural Tele. Serv. Co.</i> 499 U.S. 340 (1991)	1, 5, 14, 15
<i>Gates Rubber Co. v. Bando Chem. Indus., Ltd.</i> 9 F.3d 823 (10th Cir. 1993)	3
<i>Incredible Techs., Inc. v. Virtual Techs., Inc.</i> 400 F.3d 1007 (7th Cir. 2005)	12
<i>Johnson Controls, Inc. v. Phoenix Control Sys., Inc.</i> 886 F.2d 1173 (9th Cir. 1989)	6
<i>Johnson v. Gordon</i> 409 F.3d 12 (1st Cir. 2005)	11, 15
<i>Lotus v. Borland</i> 49 F.3d 807 (1st Cir. 1995), <i>aff'd</i> by an evenly divided court, 516 U.S. 233 (1996)	<i>passim</i>
<i>Mattel, Inc. v. MGA Entm't, Inc.</i> 616 F.3d 904 (9th Cir. 2010)	15
<i>Matthew Bender & Co., Inc. v. West Pub. Co.</i> 158 F.3d 674 (2d Cir. 1998)	4
<i>Mitel, Inc. v. Iqtel, Inc.</i> 124 F.3d 1366 (10th Cir. 1997)	9
<i>Newton v. Diamond</i> 388 F.3d 1189 (9th Cir. 2004)	13, 14

TABLE OF AUTHORITIES
(cont'd)

	<u>Page(s)</u>
<i>Northwestern Nat'l Ins. Co. v. Baltes</i> 15 F.3d 660 (7th Cir. 1994)	11
<i>Sega Enters. Ltd. v. Accolade, Inc.</i> 977 F.2d 1510 (9th Cir. 1993)	<i>passim</i>
<i>Sony Computer Entm't, Inc. v. Connectix Corp.</i> 203 F.3d 596 (9th Cir. 2000)	9, 10, 11
<i>Whelan Associates v. Jaslow Dental Lab</i> 797 F.2d 1222 (3d Cir. 1986)	6
Federal Statutes	
17 U.S.C. § 102(b)	<i>passim</i>
17 U.S.C. § 106	5
Federal Rules	
Civ. L.R. 7-3(a)	1
FED. R. CIV. P. 56(c)	14
FED. R. EVID. 201(b)	9
Federal Regulations	
37 C.F.R. § 202.1	6
37 C.F.R. § 202.3(b)(3)	12
37 C.F.R. § 202.3(b)(4)(i)(A)	12
Other Authorities	
H.R. Rep. No. 1476, 94th Cong., 2d Sess. 57 (1976), reprinted in 1976 U.S.C.C.A.N. 5659	3
Michael A. Jacobs, <i>Copyright & Compatibility</i> , 30 JURIMETRICS J. 91, 102 (1989)	4
PAUL GOLDSTEIN, GOLDSTEIN ON COPYRIGHT (3d ed. 2011)	2, 3, 7, 8

I. INTRODUCTION

Oracle’s Opposition concedes that, with the exception of portions of 12 out of several *thousand* files, it has no evidence of *any* copying by Google of any Oracle source code. Aside from those 12 files, Oracle’s copyright claim is premised on “non-literal” infringement and the incorrect notion that the Java language APIs—separate and apart from their implementations, which Google did not copy—are copyrightable.

The Java language APIs at issue are unprotectable methods of operation, as Sun (now Oracle) itself told Congress in 1994. And even if there was any creative expression in those APIs, that expression has necessarily merged into the ideas represented by the APIs or is unprotectable *scenes a faire*, or its use by Google is a fair use as a matter of law.

Oracle’s argument is largely based on the assertions that designing good APIs is difficult; that designing a highly *functional* set of APIs requires careful choices; and that given the alleged complexity of its APIs, they must contain some creative expression. But the Supreme Court rejected this type of “sweat of the brow” approach to copyrightability over twenty years ago. *Feist Pubs., Inc. v. Rural Tele. Serv. Co.*, 499 U.S. 340 (1991). An extensive effort can no more transform methods of operation into copyrightable subject matter than industrious collection can turn facts into creative expression.

Sun’s CTO was correct. Functional interfaces are not copyrightable. The Court should grant Google’s motion for summary judgment on Oracle’s copyright claim.

II. ARGUMENT

A. The Court should ignore or strike Oracle’s improper objections.

“Any evidentiary and procedural objections to the motion must be contained within the [opposition] brief or memorandum.” CIV. L.R. 7-3(a). Oracle filed 8 pages of objections in addition to a 25-page opposition. The Court should reject Oracle’s improper objections.

B. Oracle must prove “virtual identity” between the works, which it cannot do.

Where “the range of protectable and unauthorized expression is [at most] narrow, the appropriate standard for illicit copying is virtual identity.” *Apple Computer, Inc. v. Microsoft Corp.*, 35 F.3d 1435, 1439 (9th Cir. 1994). The Java language APIs are not protected at all, and

the remaining allegedly copied expression is *de minimis*. Judged against the “virtual identity” standard—or even the “substantially similar” standard—there is no copyright infringement.

C. The Java language APIs are methods of operation.

The Java language APIs are the means by which developers can access the functionality of the Java language libraries. They are, in a very literal sense, methods for operating portions of the only works that Oracle pleaded in its Amended Complaint, namely versions 1.4 and 5.0 of the Java 2 Standard Edition platform¹ (the “Asserted Works”). The APIs define the precise statements that must used for developers to access the functionalities in the Java language libraries. The APIs are thus unprotectable methods of operating the Java language libraries. 17 U.S.C. § 102(b); *see also* PAUL GOLDSTEIN, GOLDSTEIN ON COPYRIGHT § 10.5.1 (3d ed. 2011) (courts have “categorically excluded copyright protection for interface specifications”).²

1. Oracle repeatedly mischaracterizes the APIs in an attempt to show creative expression where there is none.

Oracle repeatedly mischaracterizes the APIs, implying that they contain creative expression. Oracle is wrong.

According to Oracle, the APIs “tell” how to use the libraries. Opp. at 2:27. Oracle is wrong. The APIs do not “tell” how to use the libraries, they are the *means by which one uses* the libraries; the *documentation*³ for the APIs “tells” how to use the libraries.

According to Oracle, the APIs are the “blueprint” for the libraries. *Id.* at 3:5-6. Oracle is wrong. Unlike a blueprint, which states in detail how to build a structure, the APIs merely define the functions that the *libraries* implement. The proper analogy would be to a summary building

¹ In addition to the APIs at issue, the Asserted Works include significant additional elements (such as the Java platform virtual machine and development tools) that are not the subject of Oracle’s copyright claim.

² Oracle also claims that Google’s *implementation* (i.e., the collection of Java language libraries) is a derivative work of Oracle’s APIs, *see* Opp. at 6:5-9, even though it concedes that except for portions of 12 files there is no evidence of literal code copying. This type of bootstrapping is unsupportable. Copying an idea, fact or method of operation cannot be made actionable by alleging that the non-infringing code implementing these unprotectable elements is a derivative work. If it were, the non-infringing competitors would not have prevailed in *Apple v. Microsoft*, *Sega v. Accolade*, *Sony v. Connectix* or *Lotus v. Borland*.

³ Oracle’s claim that Google’s documentation infringes is addressed separately. *See* Part II.G, *infra*.

1 plan that identifies the rooms that are to be included and other functional requirements, but does
 2 not describe implementation details. As Oracle’s expert acknowledges, the APIs are an
 3 “abstraction.” Mitchell Decl., Ex. 2 ¶ 23.

4 According to Oracle, the APIs “describe” the fields and methods in the library classes.
 5 *Id.* at 3:11. Oracle is wrong. The *documentation* might “describe” the fields and methods, but
 6 an API merely identifies the fields and methods that are included.

7 In short, the APIs are not the libraries themselves, and they do not “describe” or “tell”
 8 how to operate the libraries. Instead, the APIs are the *methods of operating* the libraries. And,
 9 by definition, methods of operation are not creative expression within the meaning of the
 10 Copyright Act. *See* 17 U.S.C. § 102(b).⁴

11 **2. Oracle’s examples of “creativity” in the design of APIs are examples of form**
 12 **following function, and thus militate *against* copyrightability.**

13 Oracle argues that the design of good APIs requires skill and creativity, and that as a
 14 result its APIs are copyrightable. But that is not the law: “Original and creative *ideas* . . . are not
 15 copyrightable . . .” *ATC Distribution Group, Inc. v. Whatever It Takes Transmissions & Parts,*
 16 *Inc.*, 402 F.3d 700, 707 (6th Cir. 2005) (citing 17 U.S.C. § 102(b)). Moreover, functional aspects
 17 of a work are not copyrightable. *Sega Enters. Ltd. v. Accolade, Inc.*, 977 F.2d 1510, 1524 (9th
 18 Cir. 1993) (citing 17 U.S.C. § 102(b)). The functional and factual aspects of a work may be
 19 copied, “as may those expressive elements of the work that ‘must necessarily be used as incident
 20 to’ expression of the underlying ideas, functional concepts, or facts.” *Id.* (quoting *Baker v.*
 21 *Selden*, 101 U.S. 99, 104 (1879)). Consistent with these core copyright principles, a work that is
 22 “largely functional” receives only weak protection under the Copyright Act. 977 F.2d at 1527.

23 The aspects of API design that Oracle points to in support of its claim of creativity all
 24 focus on enhancing the *functionality* of APIs.⁵ Oracle argues that well-designed APIs are “easier

25 ⁴ *See also* H.R. Rep. No. 1476, 94th Cong., 2d Sess. 57 (1976), reprinted in 1976
 26 U.S.C.C.A.N. 5659, 5670 (“the expression adopted by the programmer is the copyrightable
 27 element in a computer program, and . . . the actual processes or methods embodied in the
 28 program are not within the scope of the copyright law”). At least one court has suggested that
 system architecture is one of the most common places where uncopyrightable processes will be
 found. *See Gates Rubber Co. v. Bando Chem. Indus., Ltd.*, 9 F.3d 823, 837 (10th Cir. 1993).

⁵ The testimony from Google witnesses is not to the contrary. Some engineers prefer

to learn and use” (Opp. at 3:23-24), and “almost disappear from sight” (*id.* at 4:1, quoting Swoopes Decl., Ex. 21). Oracle argues, in effect, that form follows function in well-designed APIs. But designing for *functionality* is the very antithesis of creative *expression*.⁶

a. The selection of elements to include in a set of APIs is not copyrightable.

Oracle argues that its selection of API elements is protected. Opp. at 12:23-25 (“If the designer includes too little, the developers will not have the tools and flexibility they prefer. If the designer includes too much, the APIs become overwhelming and difficult to use.”). These arguments are immaterial because they do not make the APIs protectable. Determining what features are “must have” features may well require creative insights into the needs of programmers. Determining the point at which the cost of increasing difficulty of use outweighs the benefits of increased options may require refined judgment calls. But the feature set that results from this process is not a *creative expression*. Instead, it is a designer’s best guess at the optimal feature set for a product. Whether that notion is called an idea, a system, or a method of operation, it is not protected by copyright. *See* 17 U.S.C. § 102(b); *see also* *Matthew Bender & Co., Inc. v. West Pub. Co.*, 158 F.3d 674, 682 (2d Cir. 1998) (finding “[t]he creative spark is missing where . . . the author made obvious, garden-variety, or routine selections . . .”).

Oracle’s theory would lead to absurd results. If feature sets in software were copyrightable, then competitors could not include the same feature set in their software. Moreover, Oracle concedes that the Android class libraries do not include all of the Oracle API elements, and that Android includes many API elements that the Oracle APIs lack. *See* Opp. at 20:27-21:1. Thus, under Oracle’s theory, merely having a *partially overlapping* feature set

designing APIs over implementing, but that does not mean that APIs are more creatively expressive than the underlying code. Reply Astrachan Decl., Ex. 4 ¶ 32. Instead, it means that some engineers prefer coming up with ideas, while others prefer the different challenges that come from writing code in different ways to implement an idea. *Id.*

⁶ Industry practice has long reflected that APIs are uncopyrightable. For example, while AT&T required that vendors using its Unix *implementation* take a license, it understood that the specifications for the UNIX *interfaces* were not protected by copyright. *See* Michael A. Jacobs, *Copyright & Compatibility*, 30 JURIMETRICS J. 91, 102 (1989). Similarly, vendors of IBM-compatible personal computers relied on the unprotectability of interfaces in developing noninfringing alternatives to IBM’s BIOS. *See id.* Both Sun and Oracle have repeatedly relied on this principle. *See id.* at 100; Astrachan Decl., Ex. 1 ¶¶ 62-86.

1 would constitute infringement. Such a result would allow developers to misuse copyright to
 2 claim monopoly control over entire product areas. This is precisely what the idea-expression
 3 dichotomy, on which copyright law is premised, is designed to prevent.

4 Because the Java language APIs and their elements are functional, Google (and others)
 5 could have freely implemented all of the Java language APIs. But those elements are functional
 6 (and thus unprotectable) regardless of whether Google implemented all of them, or only some.
 7 Google's decision to implement only a subset of the APIs cannot transform those APIs into
 8 protectable expression. Similarly, the presence in Android of additional APIs that are not part of
 9 the Java platform cannot change the unprotectable nature of the Java language APIs, and does
 10 nothing to detract from Google's non-infringement argument. The Borland spreadsheet, for
 11 example, included commands not present in the Lotus 1-2-3 spreadsheet. *See Lotus v. Borland*,
 12 49 F.3d 807, 810 (1st Cir. 1995), *aff'd by an evenly divided court*, 516 U.S. 233 (1996).⁷

13 **b. The structure for a set of APIs is not copyrightable.**

14 Oracle also argues that "designing the appropriate structure" for a set of APIs results in
 15 creative expression. Opp. at 12:17. Oracle claims that the structure "is critical to the
 16 programmer," and that, depending on the structure chosen, an API can be "your organization's
 17 biggest asset" or can "even turn your users toward your competitor." *Id.* at 13:6-19. These
 18 arguments are irrelevant to copyrightability. Not everything of value or competitive advantage is
 19 protected by copyright. Copyright protection is limited to original and creative expression of the
 20 types eligible for protection under the Copyright Act.

21 The *structure* chosen by the designers of Oracle's APIs is functional, not creative
 22 expression.⁸ To make use of the APIs, developers must know, among other things, the package,

23 ⁷ Oracle's claims that Google "fragmented" the Java APIs and that copyright should be
 24 used to enforce its "compatibility" licensing goals, *see, e.g.*, Opp. at 8:9-15, 21:9-16, are likewise
 25 irrelevant. The Copyright Act does not protect against fragmentation or permit only
 26 "compatible" uses of unprotected elements. *See* 17 U.S.C. §§ 102(b), 106. Instead, everyone is
 27 allowed "to build freely upon the ideas and information conveyed by a work." *Feist*, 499 U.S. at
 28 350. Google is entitled to use as much or little as it wishes of Oracle's unprotectable methods of
 operation.

⁸ A phone book could be organized alphabetically by last or first name, or geographically
 by street. The fact that there are different functional ways that data can be organized does not
 make any of those functional choices copyrightable. *See Feist*, 499 U.S. at 362-63.

class, and subclass to which the methods belong; the parameters the methods take; the order of the parameters; the fields the data structures use; and how those fields are arranged. All of those features are necessary elements of the methods by which developers access the functionality in the libraries. Astrachan Decl., Ex. 1 ¶¶ 129-33; Reply Astrachan Decl., Ex. 4 ¶ 26.

c. The names of the elements in the APIs are not copyrightable.

Oracle’s opposition ignores the fundamental copyright principle that the names of the APIs and their elements are not copyrightable: “Words and short phrases such as *names*, *titles*, and slogans” are not copyrightable. 37 C.F.R. § 202.1 (emphasis added). There is no prohibition against using unprotected material, and that tautology is no less true if one uses many unprotected elements rather than just a few. Oracle’s fallback position—that the selection, coordination, and arrangement of unprotected elements can itself be expressive—fares no better, because the selection and structure of the APIs are not expressive, as explained above. *See also Lotus*, 49 F.3d at 816 (even if a software developer makes “some expressive choices” in choosing and arranging the menu items in its user interface, “that expression is not copyrightable because it is part of [the program’s] ‘method of operation.’”).

3. Oracle improperly relies on dated law and other inapposite authorities.

Oracle’s authorities do not support its position. The Ninth Circuit’s pre-*Feist* decision *Johnson Controls, Inc. v. Phoenix Control Sys., Inc.*, 886 F.2d 1173 (9th Cir. 1989), did not address software interfaces. Moreover, the *Johnson Controls* court’s statement that non-literal elements may be copyrightable where they are expression rather than ideas, *id.* at 1175, is not relevant here, because Oracle’s APIs are methods of operation, and thus by definition are not expression. 17 U.S.C. § 102(b). In addition, the analysis in *Johnson Controls* parallels the Third Circuit’s pre-*Feist* approach in *Whelan Associates v. Jaslow Dental Lab*, 797 F.2d 1222 (3d Cir. 1986), which, in the Ninth Circuit’s words, has been “widely—and soundly—criticized as simplistic and overbroad.” *Sega*, 977 F.2d at 1525-26. Oracle, in effect, invites the Court to repeat the *Whelan* court’s error of defining the “idea” of a software program unduly narrowly and then wrongly assuming that everything else must be creative expression.

The CONTU Report also supports Google’s position, not Oracle’s. While the CONTU

Report correctly states that the “written rules” of a game can be copyrighted even though they describe the process by which the game was played, CONTU Report at 21, this only means that the *prose* one uses to *explain* the mechanics of game may be copyrightable, even though the game mechanics themselves are not. *See* GOLDSTEIN ON COPYRIGHT § 2.15.1.3.⁹ As the CONTU Report explained, “[t]hat the words of a program are used ultimately in the *implementation of a process* should in no way affect their copyrightability.” CONTU Report at 21 (emphasis added). But while the *implementation* of a process (such as the *implementation* of an API) is potentially copyrightable, the process itself (like the abstract rules of a game, or an API itself) is not. 17 U.S.C. § 102(b).

Nor does *Apple Computer, Inc. v. Formula, Int’l, Inc.*, 725 F.2d 521 (9th Cir. 1984), another pre-*Feist* case, help Oracle. In that case, the Ninth Circuit rejected Formula’s argument that operating systems, which the court noted were “programs that are designed to manage the computer system,” are unprotectable methods of operation. *Id.* at 523-25. Formula conceded that the *code* it used was substantially similar to Apple’s code, *see id.* at 522-23, so there was no issue whether Formula was entitled to write code, not substantially similar to Apple’s code, that independently implemented the *interfaces* to Apple’s operating system. Apple sought “to copyright only its particular set of instructions, not the underlying computer process.” *Id.* at 525. The issue in this case—not addressed in *Apple v. Formula*—is whether the *interfaces* for accessing the Java language libraries are protected.¹⁰ And, for the reasons expressed herein, those interfaces are unprotected methods of operation.

Oracle also mistakenly claims that in *Brown Bag Software v. Symantec Corp.*, 960 F.2d 1465 (9th Cir. 1992), the Ninth Circuit “suggested” that screens, menus and keystrokes might be copyrightable. What the Ninth Circuit actually observed is that screens, menus and keystrokes might be suitable components for analytic dissection in the “extrinsic” prong of the test for

⁹ *But see Allen v. Academic Games League of America, Inc.*, 89 F.3d 614, 617-18 (9th Cir. 1996) (applying merger doctrine and denying protection even to expression in game manuals).

¹⁰ Indeed, Oracle’s expert acknowledges the distinction between the expression in implementing code and the ideas of the APIs themselves when he states that the source code that *implements* the APIs is the “written expression” of the API. Mitchell Decl., Ex. 2 ¶ 55.

substantial similarity. *Id.* at 1477. Analytic dissection is used to determine whether substantially similar elements are the result of protected or unprotected expression. *Id.* at 1475-76. The court’s reference to screens, menus and keystrokes as suitable components for analytic dissection does not suggest that those components are protectable, as *Apple v. Microsoft* made clear. Consistent with *Apple v. Microsoft*, the APIs and their selection and structure must be excluded from the infringement analysis because they are unprotectable. *See supra*, Part II.C.2.¹¹

D. Any creative expression in the Java language APIs has merged into their underlying ideas, or is unprotected *scenes a faire*.

To implement APIs that are compatible with the Java language APIs, Google had no choice but to replicate the precise details of the interfaces, just as GNU and Apache did with the GNU Classpath and Apache Harmony projects. *See Mot.* at 8:21-9:3.¹² Nonetheless, citing *Control Data Sys., Inc. v. Infoware, Inc.*, 903 F. Supp. 1316, 1323 (D. Minn. 1995), Oracle argues that because *Sun* had many options available when it designed the APIs, the particular choices *Sun* made do not merge into the ideas underlying the APIs. *Opp.* at 16:24-28. Courts in the Ninth Circuit, however, have rejected this approach. *See* GOLDSTEIN ON COPYRIGHT § 2.3.2.1 (observing that courts in the Ninth Circuit have held that “it is the range of expressive choice that existed at the time the competing product was created—not the range of expression that existed at the time the copyrighted work was created—that controls”).

For example, in *Sega*, the Ninth Circuit noted that even specific computer code can be copied if it is the only means of accomplishing a task. 977 F.2d at 1524. At issue was code *Sega* had written and by which *Sega*-authorized games “unlocked” its Genesis gaming console. *Id.* at 1524 n.7. The court did not focus on whether *the plaintiff* had more than one option available to it when writing the “unlock” code, as it likely did. The court focused instead on whether *the defendant* at the time of the alleged infringement had alternatives to using the *Sega* code. *Id.*

¹¹ Oracle also relies on *Engineering Dynamics, Inc. v. Structural Software, Inc.*, 26 F.3d 1335, 1345-46 (5th Cir. 1994), which followed the district court reasoning in *Lotus* that was reversed by the First Circuit in 1995; and *Autoskill Inc. v. National Educational Support Systems, Inc.*, 994 F.2d 1476 (10th Cir. 1993), and *CMAX/Cleveland, Inc. v. UCR, Inc.*, 804 F. Supp. 337, 355 (M.D. Ga. 1992), both of which pre-date *Lotus*, *Mitel*, and *Apple v. Microsoft*.

¹² Oracle does not dispute that it has not accused the GNU Classpath or Apache Harmony implementations of these same APIs of infringing its copyrights.

1 Concluding that there were no such alternatives, the court rejected the argument that copying the
2 unlock code infringed Sega's copyright. *Id.*

3 Oracle does not seriously dispute that to implement APIs that interoperate with the Java
4 language APIs, Google had to adopt the structure and organization of those APIs. *See* Opp. at
5 17:1-3.¹³ Given the lack of choices available to Google, any arguable expression in the Java
6 language APIs merges with the idea of providing APIs that are interoperable with the Java
7 language APIs. *See Lotus*, 49 F.3d at 818 ("In the context of methods of operation" using
8 unprotectable ideas "requires the use of the precise method of operation already employed").¹⁴

9 **E. Google's fair use case is *stronger* than that of *Sony v. Connectix*, in which the Ninth**
10 **Circuit held that Connectix's use was fair.**

11 In *Sony Computer Entm't, Inc. v. Connectix Corp.*, 203 F.3d 596 (9th Cir. 2000), the
12 Ninth Circuit held that the repeated intermediate copying of the entire code for Sony's
13 PlayStation BIOS was fair because the end product, Connectix's Virtual Game Station ("VGS"),
14 copied only unprotected elements of the BIOS. Among other things, VGS copied the "entry
15 points" (i.e., the interface) into the BIOS and implemented 137 of the 242 functions supported by
16 the BIOS. *See* Reply Kwun Decl., Ex. GG (Connectix's Opening Appellate Brief) at 13, 18.¹⁵
17 Connectix deduced the required functionality by examining the parameters sent to the BIOS and
18 the information returned by the BIOS. *See id.* at 13. In short, Connectix copied the APIs for
19 over half of the functions in the Sony PlayStation BIOS. Moreover, during the development of
20 VGS, Connectix repeatedly copied the entirety of the BIOS—i.e., Sony's *implementations* as

21 ¹³ Oracle suggests, without any supporting citation, that implementing the APIs at issue was
22 not necessary to ensure compatibility with "Java." Opp. at 17:18-19. But Oracle nowhere
23 disputes that to support use of the Java programming language (which Sun long encouraged and
24 over which Oracle claims no copyright) and provide interoperability with the well-known
25 constructs *in the Java language APIs*, Google had to make use of the names, organization and
26 structure of those APIs.

27 ¹⁴ Oracle criticizes *Lotus* by noting that the Tenth Circuit took a different approach in *Mitel*,
28 *Inc. v. Iqtel, Inc.*, 124 F.3d 1366 (10th Cir. 1997). Opp. at 11:6-12. *Mitel* relied on *scenes a*
faire, 124 F.3d at 1374-76, while *Lotus* relied on section 102(b), 49 F.3d at 815. However, under
either approach, Oracle's APIs are not protectable.

¹⁵ Google requests that the Court take judicial notice of Connectix's Opening Appellate
brief for the purpose of clarifying the facts and issues that were before the Ninth Circuit in that
case. *See* FED. R. EVID. 201(b).

1 well. *See* 203 F.3d at 605.

2 Google, like Connectix, has developed an end product that includes implementations of a
3 subset of the plaintiff's APIs. Google, like Connectix, did so for the purpose of increasing
4 interoperability. Oracle, like Sony, accuses the defendant of competing with its product. Here,
5 however, Oracle has not accused Google of making unlawful intermediate copies of its
6 implementation of the Java language APIs. And the undisputed facts show that Java has only
7 become *more* popular since this lawsuit was filed. *Compare* Kwun Decl., Ex. A (Am. Compl.)
8 ¶ 9 (alleging over 6.5 million Java developers) *with* Reply Kwun Decl., Ex. EE at 1 (recent
9 Oracle press release touting over 9 million Java developers). Far from harming Oracle's market,
10 Android apparently "strapped another set of rockets to the [Java] community's momentum," just
11 as Sun's CEO predicted. Kwun Decl., Ex. L.

12 The undisputed facts are therefore even more strongly in Google's favor than they were
13 in Connectix's favor in *Sony*. The Ninth Circuit held, as a matter of law, that Connectix's use of
14 Sony's BIOS APIs was fair. *Sony*, 203 F.3d at 608. *A fortiori*, Google's use of the Java
15 language APIs must also be fair.

16 **F. Oracle has identified no inconsistency between Google's position and Google's**
17 **treatment of its own APIs.**

18 In a single sentence with no analysis, Oracle incorrectly claims that Google "asserts
19 copyright . . . over its own APIs." Opp. at 14:15-16. None of the three cited exhibits supports
20 this conclusion. For example, the AdSense agreement states only that *to the extent that there are*
21 intellectual property rights (including copyrights) in the "AdSense API Specifications," those
22 belong to Google. *See* Swoopes Decl., Ex. 23 at 6 (§ IV.10). Moreover, the agreement defines
23 "AdSense API Specifications" to include the "Google-supplied *implementations* . . . of AdSense
24 API," not just the APIs themselves, which is what is at issue here. *Id.*, Ex. 23 at 1 (§ I)
25 (emphasis added); *see also id.*, Ex. 24 at 4 (§ III.2) & Ex. 25 at 1.

26 Oracle also suggests that Google seeks to prevent others from modifying the Android
27 core libraries while reserving the right to implement libraries with functionality that differs from
28 that of the Java platform libraries. *See* Opp. at 8:16-21. What Oracle ignores is that the

consequence of modifying the Android core libraries is not a copyright lawsuit. Instead, devices that do not meet the Android compatibility guidelines simply “may not use the Android trademark.” Reply Kwun Decl., Ex. FF at 4. Regardless of whether the compatibility guidelines are met, “[a]nyone is welcome to use the Android source code” *Id.*

G. Oracle offers no evidence of substantial similarity between the documentation of the Android core libraries and the Java language API documentation.

The entirety of Oracle’s argument that the Android API documentation is infringing is its expert’s summary “conclusion” that Google’s and Oracle’s documentation are substantially similar, and a single cite to a side-by-side comparison of excerpts from that documentation. Opp. at 6:1-4. Its expert’s “analysis” again points to the same side-by-side comparison, and offers the following allegedly “similar” documentation:

As an illustration, the Java API specification describes the method `java.security.KeyPair.getPrivate` as follows: “Returns a reference to the private key component of this key pair.” The Android spec uses the following prose: “Returns the private key.”

Mitchell Decl., Ex. 1 ¶ 207. These two phrases are not substantially similar, and no reasonable jury could conclude otherwise. Indeed, this example only supports Google’s position: the only similarities between the two phrases are from the unprotectable *facts* about the `java.security.KeyPair.getPrivate` method that they are both describing.

The Court is under no obligation to go beyond the single illustrative example that Oracle’s expert has offered. *Cf. Northwestern Nat’l Ins. Co. v. Baltes*, 15 F.3d 660, 662-63 (7th Cir. 1994) (“District judges are not archaeologists.”). However, if the Court were inclined to conduct its own comparison of the two respective sets of documentation without further guidance from Oracle, a brief review of the provided side-by-side comparisons demonstrates that any similarities arise only from unprotected facts about the APIs that are being documented. *See generally* Exs. E-F to Mitchell Decl., Ex. 1. The two sets of documentation are no more alike than two dictionaries that include and define the same words.

In any event, the minimal expressive content in Oracle’s documentation is entitled to no copyright protection at all. “[C]opyright law protects original expressions of ideas but it does not safeguard either the ideas themselves *or banal expressions of them.*” *Johnson v. Gordon*, 409

1 F.3d 12, 19 (1st Cir. 2005) (emphasis added) (citing *Feist*, 499 U.S. at 345-51); *Incredible*
 2 *Techs., Inc. v. Virtual Techs., Inc.*, 400 F.3d 1007, 1013 (7th Cir. 2005) (“utilitarian
 3 explanations” of a system “are not sufficiently original or creative to merit copyright protection,”
 4 or alternatively are protected “only against virtually identical copying”); *see also Lotus*, 49 F.3d
 5 at 815 n.9 (stating in *dicta* that a “strong argument” could be made that brief explanations of
 6 menu commands merge with the underlying idea of explaining the commands).

7 **H. The alleged copying in portions of 12 files out of several thousands of files is *de***
 8 ***minimis* and thus not actionable.**

9 **1. The alleged copying must be considered against the entirety of the Asserted**
 10 **Works to determine whether it is *de minimis*.**

11 Oracle argues that the Court should evaluate infringement on a file-by-file basis, but the
 12 authority it cites requires exactly the opposite. Even if a registered work includes “copyrightable
 13 elements that are *otherwise* recognizable as self-contained works,” 37 C.F.R. § 202.3(b)(4)(i)(A)
 14 (emphasis added), where the claimant files a “single application” and pays a “single registration
 15 fee,” all of those copyrightable elements “shall be considered a *single* work.” *Id.* § 202.3(b)(4)(i)
 16 (emphasis added). Oracle did not register the files separately. In view of the registrations Oracle
 17 obtained and on which it bases its infringement claim, the allegedly infringed “works” are the
 18 Java SE 1.4 and 5.0 platforms, not the individual files.

19 Moreover, the district court case on which Oracle relies, *Bean v. McDougal Littell*, 669 F.
 20 Supp. 2d 1031, 1034 (D. Ariz. 2008), discusses only the registration of a collective work. To
 21 claim copyright to “independent works in which copyright is being claimed and which appear
 22 within a larger work or within a collection of works,” the claimant must list the titles of all such
 23 independent works on a continuation sheet as part of the registration application. 37 C.F.R.
 24 § 202.3(b)(3). Oracle’s registrations do not list any individual files, *see Kwun Decl.*, Ex. B, nor
 25 does Oracle provide any evidence that the individual files are anything but code *fragments* that
 26 are not recognizable as self-contained works.

27 Oracle argues that comparing infringement only to the Asserted Works as a whole would
 28 lead to unjust results. This is incorrect for two reasons. First, Oracle, not Google, chose to
 register its works in the manner it did and to assert the specific copyright registrations that it has

1 asserted. Second, Oracle is wrong to claim that this allows developers to “steal files at will from
 2 large software programs.” Opp. at 24:1-2. The *de minimis* defense applies only where the
 3 allegedly copied material is *both* quantitatively and qualitatively insignificant.¹⁶

4 The law requires that the alleged copying must be assessed in comparison to the Asserted
 5 Works *as a whole*. *Newton v. Diamond*, 388 F.3d 1189, 1195 (9th Cir. 2004). Applying that
 6 test, the limited alleged copying identified by Oracle is, as a matter of law, *de minimis*.

7 **2. The rangeCheck function is not qualitatively or quantitatively significant.**

8 Oracle concedes that the rangeCheck function is quantitatively insignificant. Opp. at
 9 24:15-17 (function is “only nine lines long” and is part of a file with “3,180 lines of code”).
 10 Oracle nonetheless claims it is *qualitatively* significant because it is called nine times in the
 11 Arrays.java file. This is a non-sequitur that fails to comprehend the meaning of “qualitative”; it
 12 is like saying that the letter ‘e’ is qualitatively significant in the English language because it is
 13 the most commonly used letter. *See also* Reply Astrachan Decl., Ex. 3 ¶ 56.

14 The fact that a function is called repeatedly may have nothing to do with any qualitative
 15 significance, and everything to do with inefficient programming. *Id.* ¶ 57. And, in fact, that is
 16 the case here—due to limitations in the Java programming language, the author of Arrays.java
 17 was forced to call upon the rangeCheck code multiple times. *Id.* ¶ 58. “Thus the metric of
 18 number of calls is not a measure of the importance of rangeCheck, but rather of the inadequacies
 19 imposed by the Java language.” *Id.*

20 Oracle’s only attempt to support its claim of qualitative significance is its expert’s
 21 *speculation* that there “may” be “more to it than meets the eye” to the nine lines of the
 22 rangeCheck function. Mitchell Decl., Ex. 2 ¶ 86. Oracle’s expert *speculates* that the order of the
 23 three error checks performed *may* be significant. *Id.* But he offers no explanation or evidence in
 24 support of this speculation, opining only that it is “possible” that “some amount of trial and error
 25 went into figuring out how to arrange the tests in this code so that the most informative error

26
 27 ¹⁶ If infringement were considered on a file-by-file basis—and, for the reasons stated, that
 28 would be error—Oracle’s damages and injunctive relief theories would need to be limited to the
 individual files identified by Oracle.

condition is reported.” *Id.* Unfounded speculation cannot preclude summary judgment. *See* FED. R. CIV. P. 56(c).

Finally, even were this speculation given credence—and given the lack of any evidentiary basis, it cannot be—it at best shows that any significance to the rangeCheck function derives from a *functional choice* arrived at through *sweat of the brow*. Whatever significance one might attach to the functional order of the three error checks performed, it has no qualitative significance *under copyright law*. *See Sega*, 977 F.2d at 1524; *Feist*, 499 U.S. 359-60.

3. The eight allegedly decompiled files are not qualitatively or quantitatively significant.

Oracle effectively concedes that the eight allegedly decompiled files that Google received from a contractor are test files. Although Oracle asserts that the files were not part of the “test” directories in its works, *Opp.* at 25:15-16, that does nothing to illuminate whether they *are in fact* test files. Tellingly, Oracle’s expert does not and cannot dispute that they are test files. *See Mitchell Decl.*, Ex. 2 ¶ 95.

Oracle’s expert offers the circular speculation that because these files were part of Android, they must be important. *Id.* This speculation is belied by the *fact* that Google has *removed* these eight files from its Android distribution and did not replace them with anything else. *Bornstein Decl.* ¶ 8; *Reply Bornstein Decl.* ¶ 2. Oracle offers no evidence that the files are qualitatively significant. Its expert claims that the files are more than “dummy” files, *Mitchell Decl.*, Ex. 2 ¶ 91, that the classes implemented have a “meaningful” function, *id.* ¶ 92, that some of the methods in these files include more program logic than others, *id.* ¶ 93, and that they implement certain Interfaces, *id.* ¶ 94. Even if these assertions were true, they offer no evidence that the eight files are qualitatively *significant* to the Asserted Works *as a whole*. Oracle’s expert’s analysis does nothing to bar summary judgment.

Finally, Oracle offers no evidence to rebut Google’s showing that the eight files are quantitatively *de minimis*, judged against the Asserted Works as a whole.

4. The allegedly copied comments in the remaining two files are not qualitatively or quantitatively significant.

As Google has explained, the allegedly copied comments in the remaining two files are

descriptive and functional. Mot. at 23:15; *see also* Astrachan Decl., Ex. 1 ¶¶ 172-73. Oracle does not dispute the mundane nature of the comments. *See* Opp. at 24:24-25:2. Oracle asserts only that the comments—which Google has since removed and not replaced, Bornstein Decl. ¶ 9; Reply Bornstein Decl. ¶ 3—are “significant” to the “works from which they have been taken.” *Id.* at 25:2. But the “works” Oracle refers to are the two individual files, based on Oracle’s incorrect argument about what “works” are at issue. *See id.* at 24:3-5. Oracle offers no evidence that the comments are qualitatively significant when measured against the Asserted Works as a whole. Indeed, it is questionable whether the comments include any protected expression in the first place, because “banal expressions” are not protected, even if original. *Johnson*, 409 F.3d at 19; *see also Lotus*, 49 F.3d at 815 n.9.

And, again, Oracle offers no evidence to rebut Google’s showing that the comments in the two files are quantitatively *de minimis*, judged against the Asserted Works as a whole.

5. Google’s implementation of 37 Java language API packages is irrelevant to the *de minimis* analysis.

Oracle argues that the foregoing examples of, at most, *de minimis* copying must be considered together with Google’s decision to implement 37 Java language API packages. Oracle is again wrong. The infringement analysis proceeds only after the removal of unprotected elements via analytic dissection. *Mattel, Inc. v. MGA Entm’t, Inc.*, 616 F.3d 904, 913 (9th Cir. 2010). As explained above, the APIs are not protected, and thus are not part of the infringement analysis at all. Because the alleged copying in the portions of the 12 files is, as a matter of law, quantitatively and qualitatively insignificant when compared to the Asserted Works as a whole, Oracle’s infringement claim fails.

III. CONCLUSION

For the foregoing reasons, Google respectfully requests that the Court enter summary judgment in Google’s favor on Count VIII of Oracle’s Amended Complaint.

Dated: August 29, 2011

KEKER & VAN NEST LLP

By: s/ Michael S. Kwun
MICHAEL S. KWUN
Attorneys for Defendant GOOGLE INC.